Analysing the relative importance of robot brains and bodies

Jelisavcic, Milan; Roijers, Diederik M.; Eiben, A. E.

published in
ALIFE 2018 Proceedings of the Artificial Life Conference 2018
2018

DOI (link to publisher)
10.1162/isal_a_00063

document version
Publisher's PDF, also known as Version of record

document license
Other

Analysing the Relative Importance of Robot Brains and Bodies

Milan Jelisavcic[1], Diederik M. Roijers[1] and A.E. Eiben[1]

[1]Vrije Universiteit Amsterdam, Amsterdam, The Netherlands
m.j.jelisavcic@vu.nl

Abstract

The evolution of robots, when applied to both the morphologies and the controllers, is not only a means to obtain high-quality robot designs, but also a process that results in many body-brain-fitness data points. Inspired by this perspective, in this paper we investigate the relative importance of robot bodies and brains for a good fitness. We introduce a method to isolate and quantify the effect of the bodies and brains on the quality of the robots and perform a case study. The method is general in that it is not restricted to evolutionary systems. For the case study, we use a system of modular robots, where the bodies are evolvable and the brains are evolvable and learnable. These case studies validate the usefulness of our method and deliver interesting insights into the interplay between bodies and brains in evolutionary robotics.

Introduction

Evolutionary Robotics (ER) is a field that aims to apply evolutionary computation techniques to evolve the overall design, or controllers, or both for real and simulated autonomous robots (Vargas et al., 2014). This approach is useful both for investigating the design space of robotic applications and for testing scientific hypotheses of biological mechanisms and processes (Floreano et al., 2008). In this paper, we consider ER systems where both the morphologies and the controllers are evolvable and we investigate their effect on the quality of the robot's performance. The naive version of our research question is "What is more important for good robot behaviour, a good body or a good brain?".

To address this issue empirically, we define a task (locomotion) that induces a behaviour-based quality measure (speed). Using speed as a measure of quality we study a system of simulated modular robots with possibly complex bodies that can consist of a large number of modules, have different shapes and limbs with different lengths. The naive version of our main research question is then split into two technical questions.

Q1: Can we isolate and quantify the effect of the bodies and brains on the quality of the robots?

To this end, we present a method based on swapping brains and bodies in a given set of robots. Specifically, given N robots we create $N \times N$ new robots by pairing all N bodies with all N brains and evaluate their quality (speed, in our case). This provides us with a matrix of $N \times N$ values. Performing several statistics on this matrix we obtain the numbers that shed light on the second technical research question:

Q2: How do the effects of bodies and the brains compare?

For the current study, we use two groups of robots. The first set consists of 25 randomly generated bodies with brains that result from a gait learning process in each body independently, while the second set contains the best 25 robots found by evolution. Analysing the matrices for these two groups we conclude that our brain and body swapping method can indeed separate the effects of bodies and brains and that the relative importance depends on the inspected group of robots. In particular, we find that the differences are more prominent in the random set of morphologies than in the group of evolved robots whose bodies converged by the evolutionary process.

We hope that our brain and body swapping method offers a useful tool and that our insights will inform future ER research to be able to exploit the interplay between bodies and brains with new optimisation methods.

Related Work

Evolutionary Robotics is the combination of evolutionary computing and robotics Bongard (2013); Doncieux et al. (2011); Eiben and Smith (2015); Floreano et al. (2008); Nolfi and Floreano (2000); Trianni (2008); Vargas et al. (2014); Wang et al. (2006). The field "aims to apply evolutionary computation techniques to evolve the overall design, or controllers, or both, for real and simulated autonomous robots" Vargas et al. (2014). This approach is "useful both for investigating the design space of robotic applications and for testing scientific hypotheses of biological mechanisms and processes" Floreano et al. (2008). However, as noted in Bongard (2013) "the use of meta-heuristics [i.e., evolution] sets this sub-field of robotics apart from the mainstream of robotics research", which "aims to continuously generate better behaviour for a given robot, while the long-term goal of Evolutionary Robotics is to create general, robot-generating algorithms".

The relationship between body, brain, and the environment defines the potential for an intelligent behaviour (Beer, 2008). Auerbach and Bongard (2012) made an effort to quantify and analyse the effects of environment on morphology. Also, Smithers (1995) initiated an attempt to quantify the robot behaviour. It is also worth noting that given the right conventional controller, robots can benefit from each others knowledge, however only in a population of fixed morphologies (Heinerman et al., 2016; Haasdijk et al., 2012; Matari, 1997). Here, we attempt to quantify and set the standard for a provisional set of morphologies.

Co-evolution of Bodies and Brains

The long-term vision behind the research that inspired this work, look upon a development of robotic (eco)systems that could evolve and adapt to their environment in real-time and in real-space. This indicates that the robot morphologies are evolvable, as well as their controllers – in other words, subjects to selection and reproduction mechanisms. We presume that the robotic organisms as observed are the phenotypes encoded by their genotypes. In such systems, forming the good pair between a provisional morphology and a locomotion controller showed to be of the highest importance in order to be functional.

A real-life robotic ecosystem must satisfy several requirements in order to create a functional framework (Eiben et al., 2013). The framework, called the Triangle of Life, represents an overall system architecture with three main components or stages, illustrated in Figure 1. The first stage is the creation of a new robotic organism in the Production Centre (Jelisavcic et al., 2017a). The second stage develops in the Training Centre where a newborn robot is training to acquire skills essential for its survival. The third stage is marked as a phase of maturity where the robot in question can potentially conceive a child-robot, i.e. produce a new genome by means of selection and reproduction. Out of the three stages, the second one proved to be the most challenging for the lack of universal controller that could potentially adapt to an arbitrary morphology.

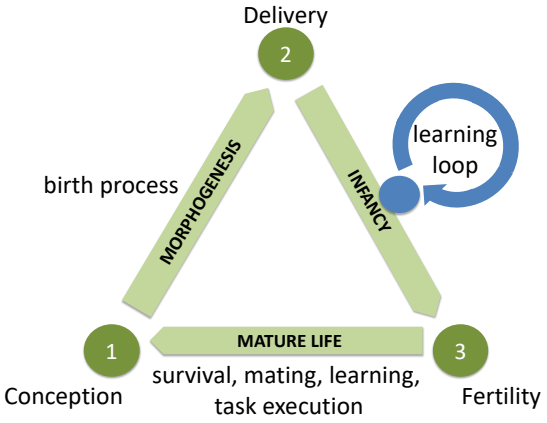


Figure 1: The Triangle of Life. The pivotal moments that span the triangle and separate the three stages are 1) *Conception*: A new genome is activated, construction of a new robot starts. 2) *Delivery*: Construction of the new robot is completed. 3) *Fertility*: The robot becomes ready to conceive offspring.

The problem of designing the universal controller has been partially solved by separating a controller's structure that is dependent on a robot's morphology and a structure that is independent. The dependent part forms a core structure in the form of a coupled CPG that governs physical actuators on a robot. The independent part defines the weights between CPG's coupled neurons based on which is formed an output pattern, thus providing the means to optimise the overall robot's behaviour.

The CPG-based controllers have been proven to perform well for modular robots. The CPG controller structure is

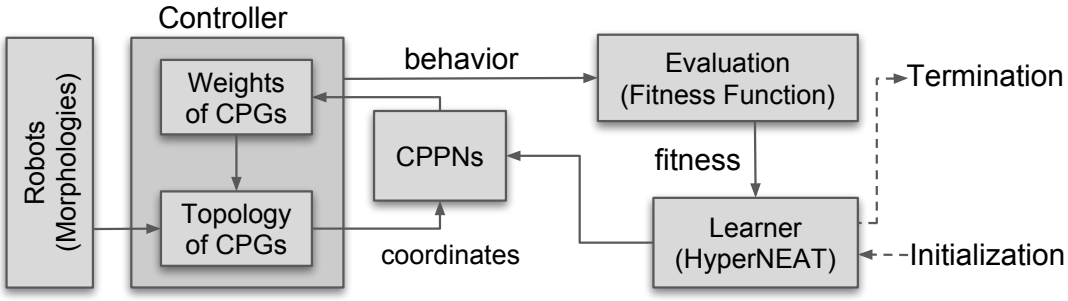


Figure 2: The overall architecture of the learning system. The working of a controller is determined by the CPG and the CPPN. The part that can be transferred between different robots is the CPPN. The CPG is strongly grounded in the morphology of a given robot.

essentially an artificial neural network with a morphology-dependent structure consisting of pairs of coupled differential oscillators logically defined for every active joint in robot's body. The main components of the CPG controllers are differential oscillators. Each oscillator is defined by two neurons that are recursively connected as shown in Fig. 3.

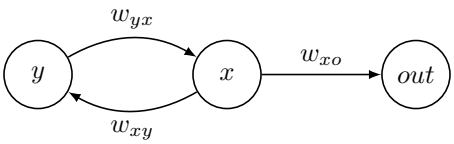


Figure 3: A differential oscillator with output node as used in the CPG controller.

These generate oscillatory patterns by calculating their activation levels x and y according to the following differential equations:

$$\dot{x} = w_{yx}y + bias_x \tag{1}$$

$$\dot{y} = w_{xy}x + bias_y \tag{2}$$

with w_{xy} and w_{yx} denoting the weights of the connections between the neurons; $bias_x$ and $bias_y$ are parameters of the neurons. If w_{yx} and w_{xy} have different signs the activation of the neurons x and y is periodic and bounded.

An oscillator's x node is connected to a linear output neuron that in turn connects to the robot's active hinge. Output neurons use the following activation function:

$$f(x) = (w_{xo} \cdot x - bias) \cdot gain. \tag{3}$$

with x the activation level from the oscillator, w_{xo} the weight of the connection between oscillator and output node and *bias* and *gain* parameters. Each active joint in the robot body is associated with an oscillator and connected to it through an output neuron that determines the joint's angle.

The oscillators of neighbouring hinges (i.e., hinges separated by a single component) are interconnected by means of weighted connections between their x neurons. This results in a chain-like neural network of differential oscillators that extends across the robot body.

We employ a lifetime learning scheme that includes HyperNEAT to evolve CPPNs, thus capacitating the Lamarckian evolution on the system as described by Jelisavcic et al. (2017b). Stanley et al. (2009) proposed HyperNEAT, an indirectly encoded evolutionary algorithm for neural networks. The idea behind HyperNEAT is to assign the nodes in a substrate neural network a location in an n-dimensional hypercube. The assigned relative positions should in some way reflect a relationship between the nodes, allowing the algorithm to exploit the geometry of the problem. The coordinates of two nodes in the hypercube are then input values for a CPPN, which outputs a value for the weight of their connection. The CPPN evolves using HyperNEAT (Stanley and Miikkulainen, 2002) so that the substrate network's performance is optimised.

The CPPN is an artificial neural network with fixed inputs and outputs structures that follow certain hyperspace coordinate system and uses coordinate values to feed the inputs. In this paper, the evolution of CPPNs follows (Stanley et al., 2009), with some modifications: it uses binary tournament selection for two parents within a species if there is more than one individual in that species. If there is only one in-

dividual in a species, the best individual of a random other species is selected as the second parent. Finally, the implementation uses elitism, transferring the best 10% of the individuals to the next population.

The CPG nodes are positioned in a three-dimensional hyperspace. Two dimensions are the relative position of the active hinges in the robot morphology as proposed by Haasdijk et al. (2010). Such *modular differentiation* allows specialisation of the active hinge's movements depending on its relative position in the robot. The hinge coordinates are obtained from a top-down view of the robot body. Thus, two coordinates of a node in the CPG controller correspond to the relative position of the active hinge it is associated with. The value of the third (z) coordinate varies according to the type of node and the kind of connection: for connections within an oscillator, X and y nodes have $z = 1$, respectively $z = -1$. For connections between neighbouring oscillators, $z = 0$.

The CPPNs have six inputs denoting the coordinates of a connection's source and target and three outputs: the weight of the connection and the bias and gain for the target node. For inter-oscillator connections (when $z = 0$), the gain and bias outputs are ignored.

The evolution of morphologies is a process determined by the performance of behaviours that is manifested on a robot. After every lifetime learning cycle, a generation of robots is evaluated for their performances. The selection process is based on two robots selection with the binary tournament. The detailed explanation of the evolutionary process of morphologies is presented in Jelisavcic et al. (2017b).

The body-brain swap method

Because we employ the architecture shown in Figure **??** our system has the property that the controller of a robot can be easily transferred to any other robot, even if they have different morphologies. To be more precise, the CPPN of robot A can be inserted into robot B and induce a controller (a CPG) that fits the morphology of robot B. Based on this property, we design a method to isolate and quantify the effects of robot brains and bodies. This method works for any given set of N robots that we view as tuples of a body and a brain. Specifically, robot i is a tuple $(body_i, brain_i)$.

Now the body-brain swap method consists of two stages. In stage 1 we create $N \times N$ new robots by pairing all N bodies with all N brains and evaluate their quality of performance $f_{ij} = f(body_i, brain_j)$ in the given environment on the given task. (In our case this is the speed of the robots.) This provides us with $N \times N$ performance values of f_{ij} that form the *swap matrix* $S_{N\times N}$ with the original body-brain pairs populating the diagonal.

In stage 2 we define the improvement for swapping a brain for a given body i as $\delta_{ij} = f_{ij} - f_{ii}$ and the improvement for swapping a body to a given brain i as $d_{ji} = f_{ji} - f_{ii}$. (Note that the improvement can be negative.) Hereby we obtain two matrices, the Δ matrix that contains the δ_{ij} values and the D matrix that contains the d_{ij} values. To see whether swapping brains or swapping bodies has a bigger effect, we compute the average of standard deviations of the improvement when averaging over all possible alternative brains (or bodies) for a given body (resp. brain). Specifically, the average improvement for a given fixed body over different brains i is $\mu_{\delta,i} = \frac{1}{N}\sum_j \delta_{ij}$ and the average improvement for a fixed brain i is $\mu_{d,i} = \frac{1}{N}\sum_j d_{ji}$. For a fixed body, the standard deviation in the improvement is then: $\sigma_{\delta,i} = \sqrt{E[(\delta_{ij} - \mu_{\delta,i})^2]}$, and analogously we define $\sigma_{d,i}$ for fixed brains.

The purpose of these matrices is to reveal and isolate the effects of introducing either a new body to an existing brain or a new brain to an existing body. Specifically, the standard deviations show how sensitive the performance of a robot is to changing its body or changing its brain. For instance, a small standard deviation $\sigma_{\delta,i}$ of changing brains in a given robot i indicates that the body is the determinative factor in the performance of this robot.

Robot test suites

We use two groups of robots for our analysis. Group 1 consists of 25 randomly generated bodies with brains that result from a gait learning process in each body independently, whereas Group 2 contains the best 25 robots found by evolution.

The robots of Group 1 are shown in Figure 4. Each robot has trained its locomotion capabilities by evolving CPPN networks using the HyperNEAT algorithm. The CPPNs generated the weights that are applied to the connections between CPG coupled neurons. The quality of a CPPN is

quantified with a locomotion speed the robot develops with a set of weights. The best performing CPPN is extracted and used for the swap matrix analysis.

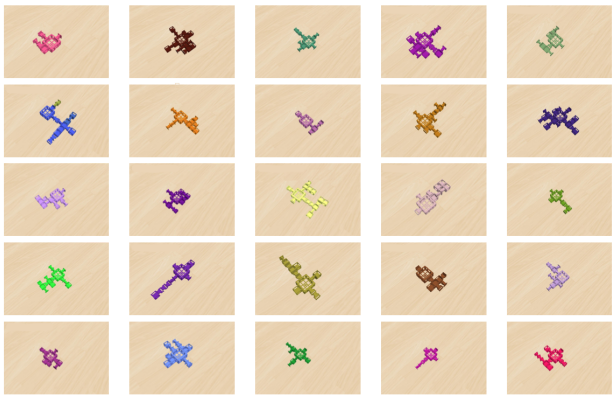


Figure 4: *G*roup 1: 25 randomly generated morphologies.

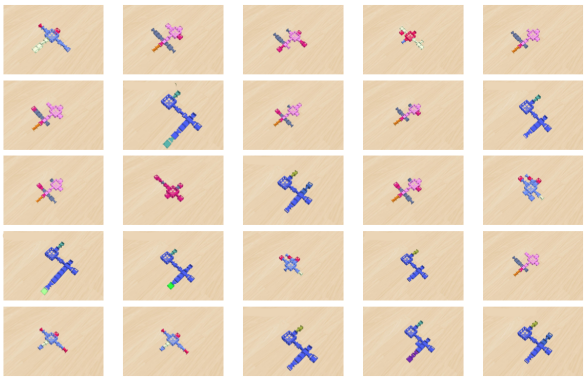


Figure 5: Group 2: The 25 best performing robots out of the 600 that were generated during the evolutionary processes in the three lineages. Note that morphologies tend to converge to similar shapes.

To create Group 2 we run Lamarckian evolution three times independently starting with a different initial population according to the algorithm described in Jelisavcic et al. (2017b). It is important to note that, in the experimental set-up from which we extracted robots for Group 2, there are two evolutionary processes. First, the evolution of *morphologies* through the cycle captured by the overall triangle in Figure 1. Second, the learning cycle within the Infancy stage is also implemented by an evolutionary algorithm since we are using HyperNEAT to develop controllers in a given robot. Clearly, the algorithm for lifetime-learning does not need to be evolutionary in general, but HyperNEAT is.

	Group 1	Group 2
Fixed Bodies (δ)	0.71	0.91
Fixed Brains (d)	0.97	0.98

Table 1: The average of the standard deviation values for the Δ matrix ("Fixed Bodies") and the D matrix ("Fixed Brains") for the 25 robots in Group 1 and Group 2.

In a system, where lifetime learning is evolutionary, Lamarckism can be simply implemented by seeding a robots initial population of controllers from that of its parents. We test the variant of seeding an offspring's population that initialises the HyperNEAT population with the best 5 CPPNs from each parent. The first generation of robots does not have a parental seed to start from, so their initial HyperNEAT population consists of randomly initialised networks only containing the input and output neurons and connections from every input to every output neuron with randomly initialised weights and neuron parameters.

We take three randomly-generated populations of 20 morphologies as a representation of an ancestral population. Each ancestral population represents a beginning of one morphological lineage. For three lineages that are tested, 10 generations are produced including an ancestral population. Out of this lineages, a subset of best-performing robots' body-brain pairs was extracted for the Group 2 analysis. With a population size of 20 and 10 generations in each of these lineages all together we generate and test 600 robots. Group 2, shown in Figure 5 is formed by the best 25 from the total 600 robots.

Analysis

In this Section, we analyse the importance of brains and bodies in the two groups of robots shown in Figure 4 and 5. According to the body-brain swap method we build the swap matrix that contains the speed f_{ij} of all body and brain combinations and the corresponding Δ and D matrices to show the effects of brain-swapping and body swapping, respectively. To avoid large tables of numbers we present these data in heat-maps using grey scales to show the numerical values in Figure 6.

We compute the average of standard deviations across all

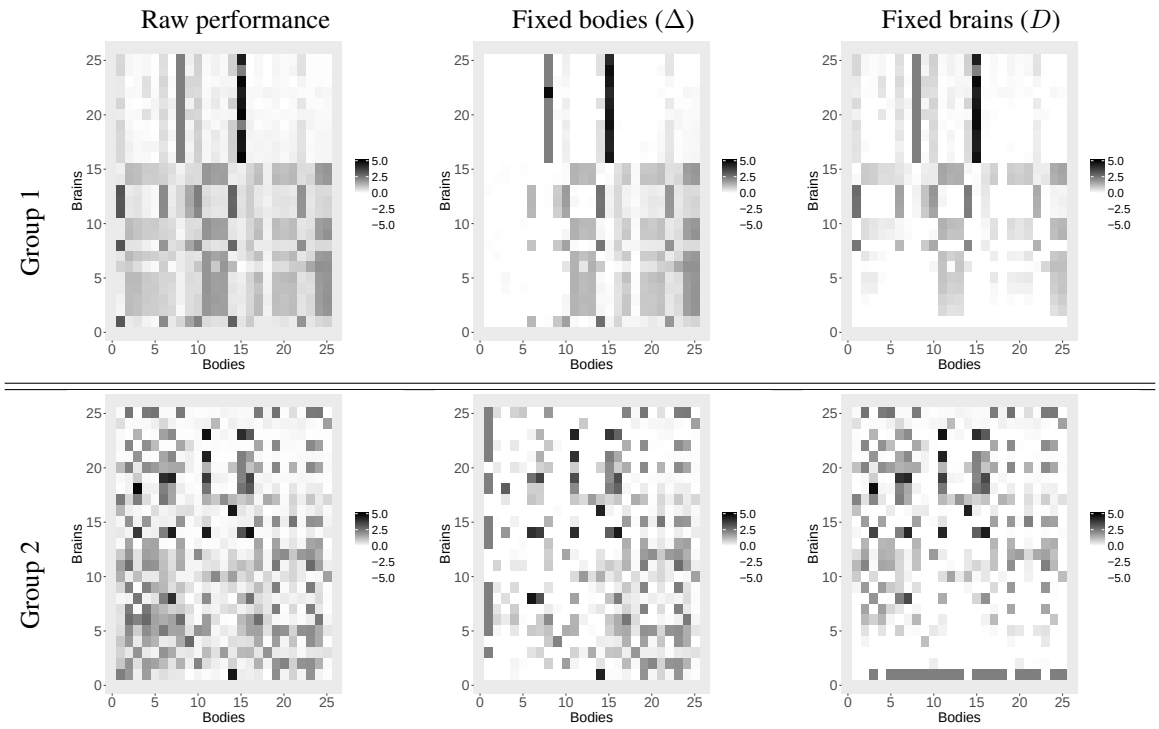


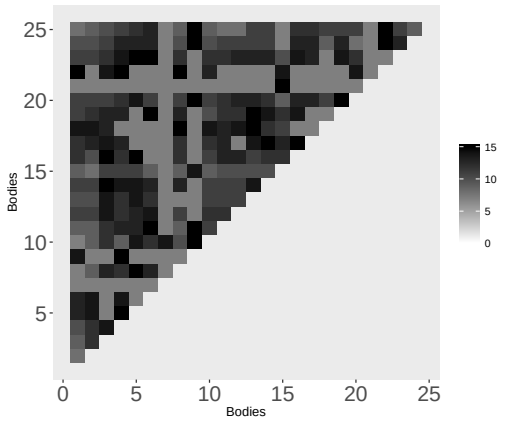
Figure 6: Heat-maps of speed (raw performance), and the calculated differences between different brains in the same body (Δ values) and different bodies to host the same brain (D values) after learning only (Group 1) and after a Lamarckian evolutionary process (Group 2).

fixed bodies and fixed brains, for the two groups of robots in Table 1. We observe that the standard deviations for swapping bodies across fixed brains are higher than the standard deviations for swapping brains across fixed bodies for both groups. However, for group 2, which is the product of a long Lamarckian evolutionary process, the difference between swapping brains and bodies is much smaller. While the absolute differences are bigger, we believe it does make sense that the differences between swapping brains and bodies become smaller, as the bodies are more similar, and the brains are further evolved in group 2. We thus conclude that for both randomly generated robots with trained brains, and evolved robots, the effect of swapping brains appears to be larger than swapping bodies.

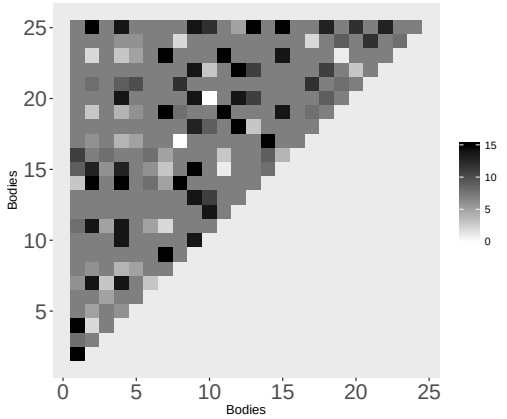
To compare the morphological differences for the two tested groups of robots, we measured individual genetic differences from robots within a group. The heat-map of each of the groups is presented in Figure 7. We can see that the population in the Group 1 is structurally far more distant from each other than in the Group 2. This validates the decision for testing random population, considering that morphologies tend to converge through evolutionary time.

To further inspect the performance of robots with brain-body swaps, we generate three heat-maps per group (Figure 6): the raw performance, f_{ij}, the improvements for swapping brains, δ_{ij}, and the improvement for swapping bodies, d_{ji}. We order the bodies on the x-axes in decreasing order of original body-brain performance, i.e., f_{ii}. We observe that again, especially for group 1, the improvements are more variable for fixed brains than for fixed bodies.

Interestingly enough, we also observe that the performances of the original body-brain pairs are not the best performing pairs in either quality of performance heat-map. Furthermore, in group 1, there exists a very high-performing body (body 15) that can achieve the highest performance (just not with its original brain). In group 2, the results of swapping are more scattered. This is probably because group 2 is the product of Lamarckian evolution. While these observations do not directly relate to our research questions, we think these observations are useful, as they



(a) Tree-edit distances for Group 1.



(b) Tree-edit distances for Group 2.

Figure 7: Morphological distances for our two groups of robots based the tree-edit distance of the Zhang and Shasha (1989) method. The robots are compared based on their mandatory attributes: *type*, *orientation*, and *slot* with whom are they connected.

seem to indicate that gains can be made by using brain-body swapping, i.e., social learning within co-evolutions of bodies and brains Haasdijk et al. (2013). The performances of the most interesting robot behaviours can be viewed on `https://www.youtube.com/playlist?list=PLwTrswqNLKxfhPWLRRvogR7j6Nmhh7EjZ`.

To conclude, we have separated the effects of bodies and brains on the quality of robot's performance with evolvable bodies as well as evolvable brains, by performing body swaps. Because of the higher variance for swapping bodies for fixed brains, we tentatively conclude that the bodies seem to have a bigger effect. This is especially the case early in the evolutionary process when there is still a lot of variation within body types (group 1), but seems to partially persist over longer periods of evolution (group 2).

Conclusions

In this paper, we introduced a new method that isolates and quantifies the effect of the bodies and brains (morphologies and controllers) on the performance of robots. We employed this method and performed an extensive data analysis to find out 'What is more important, the body or the brain?'.

Specifically, we took 25 robots that provided us with 25 bodies and 25 brains. Then we analysed the change of fitness values caused by attaching new bodies to a given brain and the change of fitness values caused by attaching new brains to a given body. Comparing the resulting standard deviations we discovered that attaching new brains to a given body resulted in smaller changes in quality than attaching new bodies to a given brain. This means that the body determined the performance of the robot more than the brain. Therefore we conclude that the bodies are more'important'. Obviously, this conclusion is only supported by the test case (the sets of robots) we use in this study and we cannot say anything about the generality of this claim.

Furthermore, we observed that the evolved body-brain pairs do not always provide the best performance. This is an interesting finding, as it indicates that there is a yet untapped potential for social sharing of controllers in evolutionary robotics. This certainly raises questions for future research.

In future work, we aim to apply the body-brain swap method to analyse populations of evolving robots for different scenarios and associated fitness functions. We aim to exploit the insights we gain from this – e.g., by applying social learning techniques – to improve the evolutionary optimisation techniques employed in evolutionary robotics.

References

Auerbach, J. E. and Bongard, J. C. (2012). On the Relationship Between Environmental and Morphological Complexity in Evolved Robots. In *Proceedings of the fourteenth international conference on Genetic and evolutionary computation conference - GECCO '12*, page 521, New York, New York, USA. ACM Press.

Beer, R. D. (2008). The Dynamics of Brain-Body-Environment Systems. In *Handbook of Cognitive Science*, pages 99–120. Elsevier, Amsterdam.

Bongard, J. C. (2013). Evolutionary robotics. *Communications of the ACM*, 56(8):74–83.

Doncieux, S., Mouret, J.-B., Bredeche, N., and Padois, V. (2011). Evolutionary robotics: Exploring new horizons. In Doncieux, S., Bredèche, N., and Mouret, J.-B., editors, *New Horizons in Evolutionary Robotics: Extended Contributions from the 2009 EvoDeRob Workshop*, volume 341 of *Studies in Computational Intelligence*, pages 3–25. Springer Berlin Heidelberg, Berlin, Heidelberg.

Eiben, A. and Smith, J. (2015). From evolutionary computation to the evolution of things. *Nature*, 521(7553):476–482.

Eiben, A. E., Bredeche, N., Hoogendoorn, M., Stradner, J., Timmis, J., Tyrrell, A., and Winfield, A. F. T. (2013). The Triangle of Life: Evolving Robots in Real-time and Real-space. In Liò, P., Miglino, O., Nicosia, G., Nolfi, S., and Pavone, M., editors, *Advances in Artificial Life, ECAL 2013*, pages 1056–1063, Taormina, Italy. MIT Press.

Floreano, D., Husbands, P., and Nolfi, S. (2008). Evolutionary robotics. In *Springer handbook of robotics*, pages 1423–1451. Springer.

Haasdijk, E., Eiben, a. E., and Winfield, A. F. T. (2013). Individual, Social and Evolutionary Adaptation in Collective Systems. *Handbook of Collective Robotics - Fundamentals and Challenges*, pages 411–469.

Haasdijk, E., Rusu, A. A., and Eiben, A. E. (2010). Hyperneat for locomotion control in modular robots. In Tempesti, G., Tyrrell, A. M., and Miller, J. F., editors, *Evolvable Systems: From Biology to Hardware*, pages 169–180, Berlin, Heidelberg. Springer Berlin Heidelberg.

Haasdijk, E., Smit, S. K., and Eiben, A. E. (2012). Exploratory analysis of an on-line evolutionary algorithm in simulated robots. *Evolutionary Intelligence*, 5(4):213–230.

Heinerman, J., Zonta, A., Haasdijk, E., and Eiben, A. E. (2016). On-line Evolution of Foraging Behaviour in a Population of Real Robots. In Di Chio, C., Brabazon, A., Di Caro, G. A., Ebner, M., Farooq, M., Fink, A., Grahl, J., Greenfield, G., Machado, P., O'Neill, M., Tarantino, E., and Urquhart, N., editors, *Applications of Evolutionary Computation*, volume 6025 of *Lecture Notes in Computer Science*, pages 198–212. Springer Berlin Heidelberg, Berlin, Heidelberg.

Jelisavcic, M., De Carlo, M., Hupkes, E., Eustratiadis, P., Orlowski, J., Haasdijk, E., E. Auerbach, J., and Eiben, A. (2017a). Real-World Evolution of Robot Morphologies: A Proof of Concept. *Artificial life*, 23(2):206–235. PMID: 28513201.

Jelisavcic, M., Kiesel, R., Glette, K., Haasdijk, E., and Eiben, A. (2017b). Analysis of lamarckian evolution in morphologically evolving robots. In *Proceedings of the European Conference on Artificial Life 2017, ECAL 2017*, volume 14, pages 214–221. MIT Press.

Matari, M. J. (1997). Learning social behavior. *Robotics and Autonomous Systems*, 20(2):191 – 204. Practice and Future of Autonomous Agents.

Nolfi, S. and Floreano, D. (2000). *Evolutionary robotics: The biology, intelligence, and technology of self-organizing machines*. MIT press.

Smithers, T. (1995). On quantitative performance measures of robot behaviour. *Robotics and Autonomous Systems*, 15(1):107 – 133. The Biology and Technology of Intelligent Autonomous Agents.

Stanley, K. O., D'Ambrosio, D. B., and Gauci, J. (2009). A hypercube-based encoding for evolving large-scale neural networks. *Artificial Life*, 15(2):185–212.

Stanley, K. O. and Miikkulainen, R. (2002). Evolving neural networks through augmenting topologies. *Evolutionary computation*, 10(2):99–127.

Trianni, V. (2008). *Evolutionary swarm robotics: evolving self-organising behaviours in groups of autonomous robots*, volume 108. Springer.

Vargas, P. A., Di Paolo, E. A., Harvey, I., and Husbands, P. (2014). *The horizons of evolutionary robotics*. MIT Press.

Wang, L., Tan, K. C., and Chew, C. M. (2006). *Evolutionary robotics: from algorithms to implementations*. World Scientific.

Zhang, K. and Shasha, D. (1989). Simple fast algorithms for the editing distance between trees and related problems. *SIAM journal on computing*, 18(6):1245–1262.